CLAIMS

1. A method, comprising:

executing a program on a computer, without the program having been compiled for profiled execution, the program being coded in an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in the binary representation of the instruction, the computer including instruction pipeline circuitry configured to execute instructions of the computer, and profile circuitry configured to detect and record, without compiler assistance for execution profiling, profile information describing a sequence of events occurring in the instruction pipeline;

during a profile-quiescent interval of execution of the program that induces events that match time-independent selection criteria of profileable events to be profiled, configuring the profile circuitry to record no profile information in response to the occurrence of profileable events;

after a triggering event is detected, the triggering event being one of a predefined class of triggering events, configuring the profile circuitry to commence a profiled execution interval and to record profile information describing every event during a profiled execution interval that matches the time-independent profileable event selection criteria induced during the profiled execution interval, including at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution;

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change taken together with a processor mode before the mode change instruction the recording continuing until a predetermined stop condition is reached;.

the recorded profile information being efficiently tailored to annotate the profiled binary code with sufficient processor mode information to resolve mode-dependency in the binary coding, and indicating contiguous ranges of sequential instructions executed during a profiled interval by low and high boundaries of the contiguous ranges, indicating the high boundary by the address of the last byte of the range, the profile information further identifying each distinct physical page of instruction text executed during the execution interval.

2. A method, comprising:

executing a program on a computer;

recording profile information concerning the execution of the program, the profile information recording of the address of the last byte of at least one instruction executed by the computer during a profiled interval of the execution.

3. The method of claim 2:

wherein the program has been compiled, without special consideration for execution profiling, into an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction;

and further comprising recording profile information describing the processor mode during the profiled execution interval, the profile information being efficiently tailored to annotate the profiled binary code with sufficient processor mode information to resolve mode-dependency in the binary coding.

4. The method of claim 2:

wherein the program has been compiled without special consideration for execution profiling;

and further comprising, commencing the profiled execution interval at the expiration of a timer, the recorded profile describing a sequence of events including every event that matches time-independent selection criteria of events to be profiled, the recording continuing until a predetermined stop condition is reached.

5. The method of claim 2:

wherein the program has been compiled without special consideration for execution profiling;

wherein the program execution induces occurrence of events that match time-independent criteria of profileable events to be profiled;

and further comprising, during a profile-quiescent interval of execution, recording no profile information in response to the occurrence of profileable events; and

commencing the profiled execution interval after a triggering event is detected, the triggering event being one of a predefined class of triggering events, the recorded profile information describing every event that matches the profileable event selection criteria induced after the triggering event, the recording continuing until a predetermined stop condition is reached.

6. The method of claim 5, wherein the triggering event is the expiration of a timer.

7. The method of claim 5:

and further comprising, without software intervention, recording for later analysis a profile entry noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

8. The method of claim 5:

wherein the program is coded in an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction;

the recorded profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution;

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction;

the recorded profile information further identifying each distinct physical page of instruction text executed during the profiled execution interval.

9. The method of claim 5:

wherein the program is coded in an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction;

and further comprising, recording profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution, and

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction.

10. The method of claim 5:

the profile information further identifying each distinct physical page of instruction text executed during the profiled execution interval.

11. The method of claim 5, wherein the program is executed on a computer having:

an instruction pipeline configured to execute instructions from a memory of the computer; and

profile circuitry configured to detect the occurrence of profileable events occurring in the instruction pipeline, and to direct the instruction pipeline to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under hardware control without software intervention.

12. The method of claim 11, wherein:

the profile information is recorded into general registers of the computer under hardware control, without software intervention, and without first storing the profile information into main memory.

13. The method of claim 5, wherein:

the profile information is recorded into general registers of the computer under hardware control, without software intervention, and without first storing the profile information into main memory.

14. The method of claim 13, the recorded profile information being efficiently tailored to identify all bytes of object code executed during the profiled execution interval, without reference to the binary code of the program.

15. The method of claim 2:

wherein the program has been compiled without special consideration for execution profiling;

and further comprising, without software intervention, recording for later analysis a profile entry noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

16. The method of claim 2:

wherein the program has been compiled, without special consideration for execution profiling, into an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction;

the recorded profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution;

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction;

the recorded profile information further identifying each distinct physical page of instruction text executed during the profiled execution interval.

17. The method of claim 2:

wherein the program has been compiled, without special consideration for execution profiling, into an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction;

and further comprising, recording profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution, and

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction.

18. The method of claim 17, wherein the divergence from sequential execution flow is consequent to recognizing or handling of an exception.

19. The method of claim 17, wherein:

an instruction of the computer, having a primary effect on the execution the computer not related to profiling, has an immediate field for an event code encoding the nature of a profileable event to be recorded in the profile information, the immediate field having no effect on computer execution except to determine the event code of the profiled event.

20. The method of claim 17, further comprising:

recording profile information that records a sequence of events of the program, the sequence including every event during the profiled execution interval that matches time-independent criteria of profileable events to be profiled.

21. The method of claim 2:

wherein the program has been compiled, without special consideration for execution profiling;

the profile information further identifying each distinct physical page of instruction text executed during the profiled execution interval.

22. The method of claim 2, the recorded profile information being efficiently tailored to identify all bytes of object code executed during the profiled execution interval, without reference to the binary code of the program.

23. The method of claim 2, wherein the recorded physical memory references include addresses of binary instructions referenced by an instruction pointer, and at least one of the

recorded instruction references records the event of a sequential execution flow across a page boundary in the address space.

24. A computer, comprising:

an instruction pipeline configured to execute instructions of the computer;

profile circuitry configured to detect, without compiler assistance for execution profiling, the occurrence of profileable events occurring in the instruction pipeline, and to direct recording of profile information representing the events, the profile information recording of the address of the last byte of at least one instruction executed by the computer during a profiled interval of the execution.

25. The computer of claim 24, wherein:

the instruction pipeline is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing a processor mode during the profiled execution interval, the profile information being efficiently tailored to annotate the profiled binary code with sufficient processor mode information to resolve mode-dependency in the binary coding.

26. The computer of claims 25, the recorded profile information further being efficiently tailored to identify all bytes of object code executed during the profiled execution interval, without reference to the binary text of the program.

27. The computer of claim 24, wherein:

during a profile-quiescent interval of execution of a program that has been compiled without special consideration for execution profiling and that induces events that match time-independent criteria of profileable events to be profiled, the profile circuitry is configured to record no profile information in response to the occurrence of profileable events; and

after a triggering event is detected, the triggering event being one of a predefined class of triggering events, the profile circuitry is configured to commence the profiled execution interval

and to record profile information describing every event that matches the profileable event selection criteria induced during the profiled execution interval, the recording continuing until a predetermined stop condition is reached.

28. The computer of claim 24:

the profile circuitry being configured to record at least one physical memory reference noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

29. The computer of claim 24, wherein:

the instruction pipeline is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution, and

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction.

30. The computer of claim 29, wherein:

the profile circuitry is configured to detect the occurrence of profileable events occurring in the instruction pipeline, and to direct the instruction pipeline to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under hardware control without software intervention.

31. The computer of claim 29, wherein:

the profile circuitry is configured to record profile information into general registers of the computer under hardware control, without software intervention, and without first storing the profile information into main memory.

32. The computer of claim 24, wherein:

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.

33. The computer of claim 24, wherein:

the instruction pipeline is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution;

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction; and

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.

34. A method, comprising:

executing a program on a computer, without the program having been compiled for profiled execution, the program being coded in an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in the binary representation of the instruction;

recording profile information describing an interval of the program's execution and processor mode during the profiled interval of the program, the profile information being efficiently tailored to annotate the profiled binary code with sufficient processor mode information to resolve mode-dependency in the binary coding.

35. The method of claim 34, the recorded profile information further being efficiently tailored to identify all bytes of object code executed during the profiled execution interval, without reference to the binary code of the program.

36. The method of claim 34:

wherein the program has been compiled without special consideration for execution profiling;

wherein the program execution induces occurrence of events that match time-independent criteria of profileable events to be profiled, the criteria including at least some physical memory references referenced by the program;

and further comprising, during a profile-quiescent interval of execution, recording no profile information in response to the occurrence of profileable events; and

commencing the profiled execution interval after a triggering event is detected, the triggering event being one of a predefined class of triggering events, the recorded profile information describing every event that matches the profileable event selection criteria induced after the triggering event, the recording continuing until a predetermined stop condition is reached.

37. The method of claim 36, wherein the triggering event is the expiration of a timer.

38. The method of claim 34:

wherein the program has been compiled without special consideration for execution profiling;

and further comprising, commencing the profiled execution interval at the expiration of a timer, the recorded profile describing a sequence of events including every event that matches time-independent selection criteria of events to be profiled, the recording continuing until a predetermined stop condition is reached.

39. The method of claim 34:

and further comprising, without software intervention, recording for later analysis a profile entry noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

40. The method of claim 34, wherein the program is executed on a computer having:

an instruction pipeline configured to execute instructions from a memory of the computer; and

profile circuitry configured to detect the occurrence of profileable events occurring in the instruction pipeline, and to direct the instruction pipeline to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under hardware control without software intervention.

41. The method of claim 34, wherein the recorded physical memory references include addresses of binary instructions referenced by an instruction pointer, and at least one of the recorded instruction references records the event of a sequential execution flow across a page boundary in the address space.

42. The method of claim 41, wherein the recorded profile information includes a record denoting a page boundary of the address space occurring within a single instruction.

43. The method of claim 41, wherein the recorded profile information includes a record denoting a page boundary between two instructions that are sequentially adjacent in the logical address space.

44. The method of claim 34, further comprising:

recording profile information recording a data-dependent change to a full/empty mask for registers of the computer.

45. The method of claim 34, wherein a profile entry describing a single profileable event explicitly describes a page offset of the location of the event, and inherits a page number of the location of the event from the immediately preceding profile entry.

46. A computer, comprising:

an instruction pipeline configured to execute instructions of the computer, coded in an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in the binary representation of the instruction;

profile circuitry configured to detect, without compiler assistance for execution profiling, the occurrence of profileable events occurring in the instruction pipeline, and to direct recording of profile information describing an interval of the program's execution and processor mode during a profiled interval of the program, the profile information being efficiently tailored to annotate the profiled binary code with sufficient processor mode information to resolve mode-dependency in the binary coding.

47. The computer of claim 46, wherein:

during a profile-quiescent interval of execution of a program that has been compiled without special consideration for execution profiling and that induces events that match time-independent criteria of profileable events to be profiled, the profile circuitry is configured to record no profile information in response to the occurrence of profileable events; and

after a triggering event is detected, the triggering event being one of a predefined class of triggering events, the profile circuitry is configured to commence the profiled execution interval and to record profile information describing every event that matches the profileable event selection criteria induced during the profiled execution interval, the recording continuing until a predetermined stop condition is reached.

48. The computer of claim 46:

the profile circuitry being configured to record at least one physical memory reference noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

49. The computer of claim 46, wherein:

the instruction pipeline is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution, and

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction.

50. The computer of claim 46, wherein:

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.

51. The computer of claim 46, wherein:

the instruction pipeline is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution;

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction; and

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.